CLAIMS:

What is claimed is:

1. A method comprising:

establishing secured communication between a client device and server device;

wherein communication is secured using, at least in part, synchronized security sequence value(s);

storing a security sequence value as a resynchronization value;

detecting at least one event desynchronizing said secured communication; and

requesting resynchronization of security sequence values, comprising sending at least a representation of said resynchronization value from said client device to said server device.

2. The method of claim 1, further comprising performing anti-replay filtering using said security sequence values.

3. The method of claim 1, wherein sending at least a representation of said resynchronization value includes embedding said resynchronization value in at least one header and/or at least one payload of a data packet.

4. The method of claim 1, wherein said storing a client resynchronization value includes periodically refreshing a stored value with a new value at a selected interval from security sequence values already used in a secured communication session.

5. A method comprising:

establishing secured communication between a client device and server device; wherein communication is secured using, at least in part, synchronized security sequence value(s);

acknowledging a client request for resynchronization, comprising sending at least a representation of said request for resynchronization and a server resynchronization value from said server device to said client device; and

reestablishing secured communication using said server resynchronization value.

6. The method of claim 5, wherein said client request for resynchronization is a client resynchronization value and said secured communication is reestablished using said client resynchronization value and said server resynchronization value.

7. The method of claim 6, wherein sending at least a representation of said client and said server resynchronization values includes embedding said client and said server resynchronization values in at least one header and/or at least one payload of a data packet that conforms to IPsec standards.

8. The method of claim 5, further comprising performing said method using a state machine in network circuitry.

9. The method of claim 5, further comprising using software to perform said method.

10. The method of claim 5, further comprising performing anti-replay filtering using said synchronized security sequence values.

11. The method of claim 5, further comprising reestablishing secured communication during a low-power state.

12. The method of claim 5, further comprising reestablishing secured communication while said first device lacks an active operating system and/or lacks an active microprocessor.

13. The method of claim 5, further comprising a machine-readable medium that provides instructions, which when executed by at least one electronic circuit, cause said at least one electronic circuit to perform operations comprising said method.

14. An apparatus, comprising;

(a) a security interface to engage in secured communication with at least one network node, wherein said security interface and said at least one network node use synchronized security sequence values at least in part to authenticate said secured communication;

(i) a recorder to store at least one security sequence value;

(ii) a desynchronization detector coupled to said security interface;

(iii) a resynchronization requester to send the stored security sequence value to at least one network node after a desynchronization; and

(iv) a verifier to receive feedback from said at least one network node;

(b) a security agent coupled to said at least one network node, comprising:

(i) a request receiver to recognize said stored security sequence value; and

(ii) an acknowledger to send said feedback from said security agent to said security interface; said feedback comprising said stored security sequence value and a node security sequence value from said network node.

15. The apparatus of claim 14, wherein stored security sequence values and node security sequence values are embedded in at least one header and/or at least one payload of a data packet that conforms to IPsec standards.

16. The apparatus of claim 14, wherein said stored security sequence value is periodically refreshed with a value at a selected interval from security sequence values already used in a secured communication session.

17. A computer network security sequence value resynchronizer, comprising:

(a) a sender having at least access to a nonvolatile random access memory;

(b) said sender to transmit a data packet containing at least in part a stored sender resynchronization value from said nonvolatile random access memory over said computer network; and

(c) an acknowledger connected to said computer network to receive said sender resynchronization value from said sender; said acknowledger returning said sender resynchronization value to said sender as security assurance.

18. The resynchronizer of claim 17, said acknowledger returning an acknowledger resynchronization value to said sender in addition to said sender resynchronization value.

19. The resynchronizer of claim 17, wherein at least one sender and at least one acknowledger are installed on any combination of server and client devices.

20. A method comprising:

establishing secured communication between a security interface and a network node, said security interface to resynchronize security sequence values between said security interface and said network node;

storing a first resynchronization value selected by said security interface; and

resynchronizing said security sequence values after a break in said secured communication, said resynchronizing further comprising:

sending said first resynchronization value from said security interface to said network node;

sending said first resynchronization value and a second resynchronization value from said network node to said security interface; and

reestablishing said secured communication using said first resynchronization value and said second resynchronization value.

21. The method of claim 20 further comprising using a security interface as a state machine in network circuitry.

22. The method of claim 20 further comprising using a security interface as a software program.

23. The method of claim 20 further comprising storing said first resynchronization value in a nonvolatile storage medium.

24. The method of claim 20 further comprising establishing secured communication using IPsec security standards.

25. The method of claim 20 further comprising resynchronizing said secured communication using said first resynchronization value to resynchronize secured data sent from said security interface and using said second resynchronization value to resychronize secured data sent from said network node.

26. The method of claim 20 further comprising resynchronizing secured communication during a low-power state.

27. The method of claim 20 further comprising resynchronizing secured communication while said network node lacks an active operating system and/or lacks an active microprocessor.

28. A method, comprising:

establishing secured communication between a server device and a client device, said secured communication using server security sequence values synchronized with client security sequence values;

storing at least one client security sequence value in nonvolatile memory as a saved client security sequence value; and

resynchronizing server and client security sequence values after a desynchronization event by sending said saved client security sequence value from said nonvolatile memory to said server device.

29. The method of claim 28, said resynchronizing further comprising returning said saved client security sequence value from said server device to said client device in a data packet with a server security sequence value.

30. The method of claim 28, said storing further comprising periodically refreshing said saved client security sequence value with a number that is greater in value than client security sequence values that have already been sent to said server device in a communication session.